

# **AR-B104B\_4P/8P, PCI104 COM**

**RS232/RS485/ Full duplex with Isolation and ESD protect**

## **User's Manual**

**Manual Rev: 1.0**

**Book Number: AR-B104B\_4P/8P-09.2.4**

## Revision

Version	Date	Author	Description
0.1	2009/1/15	Marc Chung	Draft
0.2	2009/2/3	Judy Tseng	Revise
1.0	2009/2/4	Marc Chung	Release

## **@Copyright 2008**

### **All Rights Reserved.**

Manual's first edition: January 16, 2009

For the purpose of improving reliability, design and function, the information in this document is subject to change without prior notice, which does not represent a commitment on the part of the manufacturer.

In no event will the manufacturer be liable for direct, indirect, special, incidental, or consequential damages arising out of the use or inability to use the product or documentation, even if advised of the possibility of such damages.

This document contains proprietary information protected by copyright. All rights are reserved. No part of this Manual may be reproduced by any mechanical, electronic, or other means in any form without prior written permission of the manufacturer.

### **Trademarks**

AR-B104B\_4P/8P is registered trademarks of Acrosser, IBM PC is a registered trademark of International Business Machines Corporation. Pentium is a registered trademark of Intel Technologies, Inc. Award is registered trademarks of Award Software International, Inc. Other product names mentioned herein are used for identification purposes only and may be trademarks and/or registered trademarks of their respective companies.

## Table of Contents

1.1 Specifications .....	6
1.2 Package Contents .....	7
1.3 Block Diagram .....	8
2.1 Locations (Top side).....	9
2.2 Connectors and Jumper Setting .....	11
2.2.1 COM_1_4 .....	11
2.2.2 JP1 .....	11
2.2.3 COM_5_8 .....	12
2.2.4 DB-9 Serial port pin assignment .....	12
3.1 AR-B104B Selectable Setup .....	13
4.1 Constant Definition .....	14
4.2 API Definition & Description .....	15
4.3 Sample Code : .....	16
4.4 The Function Detected 8 PORT .....	17

# 1 INTRODUCTION

Welcome to the AR-B104B\_4P/8P PCI104 COM PORT. The AR-B104B\_4P/8P is the PCI 104 compliant that compatible with 16C550 performance UART channel and supports all RS-232C transceiver mode. Maximum baud rate is to 15Mbps in asynchronous mode.

The AR-B104B\_4P/8P is a Complete Platform that totally supports 4 UART channels(AR-B104B/4P), could option to 8 channel(AR-B104B/8P)

## 1.1 Specifications

- Module Name: AR-B104B/4P or 8P RAM memory.
- Dimension: 90.2mm x 95.9mm.
- Controller: Oxford OXmPCI954.
- RS232: Isolation 1KV and ESD contact 8KV air 15KV protect.
- Function selectable: RS232/RS485/RS422 selectable by controller GPIO  
Reserve one GPIO from Oxford to identify the board is AR-B104B/4P(pull low)  
or AR-B104B/8P(pull high).
- System Interface: PCI Local bus specification V3.0.
- Connector: 2.0m.m 2x 22, Straight.
- Operation Temperature: 0 to 60 Degree C.
- Support: OS Windows XP/XPE, Linux FC6/FC7.
- **RS-232/RS-422/RS-485 type adjustable by software control tool.**

### A. Software tool requirement

1. **Need to have API. To provide to customer whom has capability to build their own application.**
2. **Need to have UI to configure 8 bit port (each), could be generate a script.**  
**Every time when OS restart, it can read the script then set parameter for each port(8 bit).**
3. **OS support in Linux(Fedora,Ubanto) and XP / XP Embedded.**
4. **Bios adjustable function 'disabled' as default, but keep in BIOS code for ODM special request utilization.**

### B. UI screen reference:

AR-B104B-4P(AR-B104B-8P) serial port configure tool

Setting protocol

COM1	<input type="radio"/> RS-232	<input type="radio"/> RS-422	<input type="radio"/> RS-485	COM5	<input type="radio"/> RS-232	<input type="radio"/> RS-422	<input type="radio"/> RS-485
COM2	<input type="radio"/> RS-232	<input type="radio"/> RS-422	<input type="radio"/> RS-485	COM6	<input type="radio"/> RS-232	<input type="radio"/> RS-422	<input type="radio"/> RS-485
COM3	<input type="radio"/> RS-232	<input type="radio"/> RS-422	<input type="radio"/> RS-485	COM7	<input type="radio"/> RS-232	<input type="radio"/> RS-422	<input type="radio"/> RS-485
COM4	<input type="radio"/> RS-232	<input type="radio"/> RS-422	<input type="radio"/> RS-485	COM8	<input type="radio"/> RS-232	<input type="radio"/> RS-422	<input type="radio"/> RS-485

Generate script: change immediately and create script for another tool to read it when next time boot up

Cancel: no change.

### C. There should be another tool to read the script to configure COM port setting.

## 1.2 Package Contents

Check if the following items are included in the package.

AR-B104B\_4P Packing List:

- AR-B104B\_4P
- 2.0mm 2x22 Pin header to 4 x DB9 cable
- 1 software utility CD
- Gift Box
- User Manual

AR-B104B\_8P Packing List:

- AR-B104B\_8P
- 2.0mm 2x22 Pin header to 4 x DB9 cable
- 2.0mm 2x22 Pin header to DB37 cable
- DB37 to 4 x DB9 cable
- 1 software utility CD
- Gift Box
- User Manual

### 1.3 Block Diagram

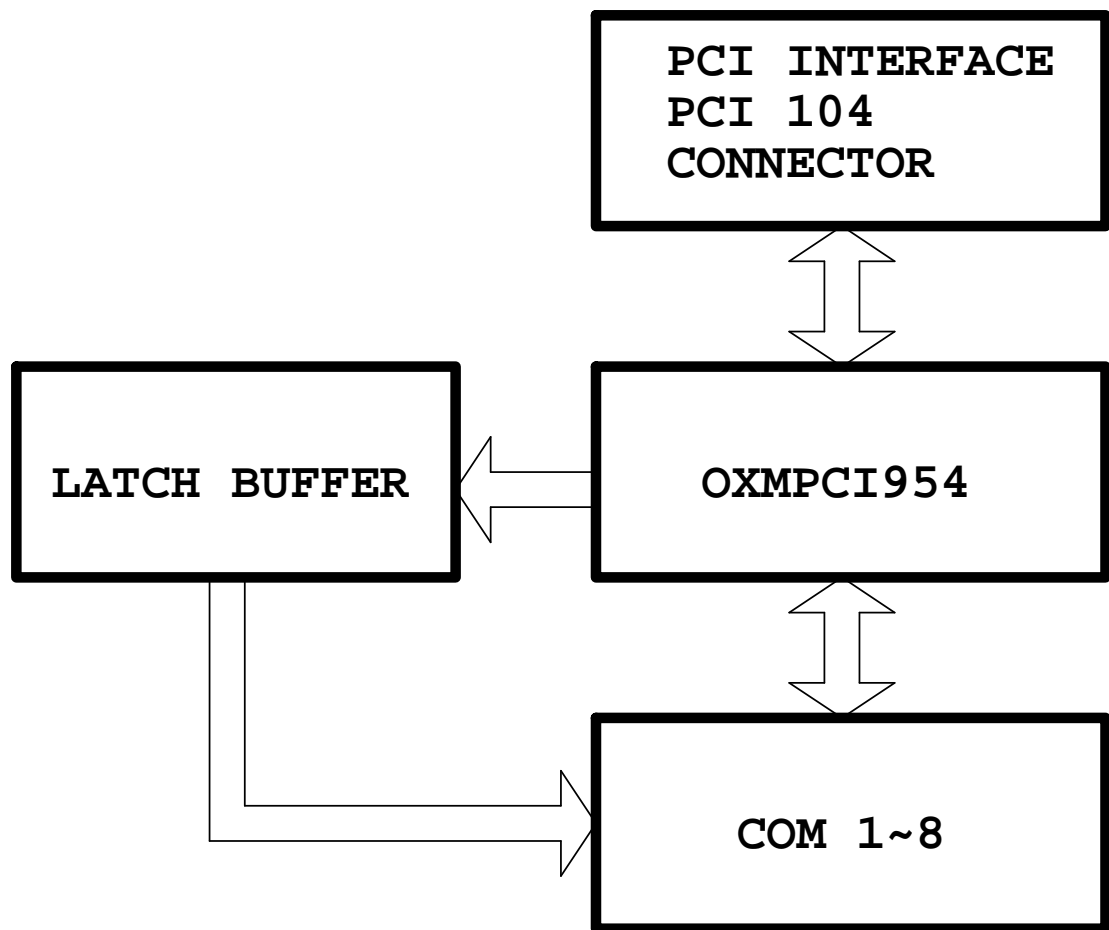


Figure 1: Block Diagram



## 2 H/W INFORMATION

This chapter describes the installation of AR-B104B\_4P/8P. At first, it shows the Function diagram and the layout of AR-B104B\_4P/8P. It then describes the unpacking information which you should be careful with, as well as the jumper/switch settings for the AR-B104B\_4P/8P configuration

### 2.1 Locations (Top side)

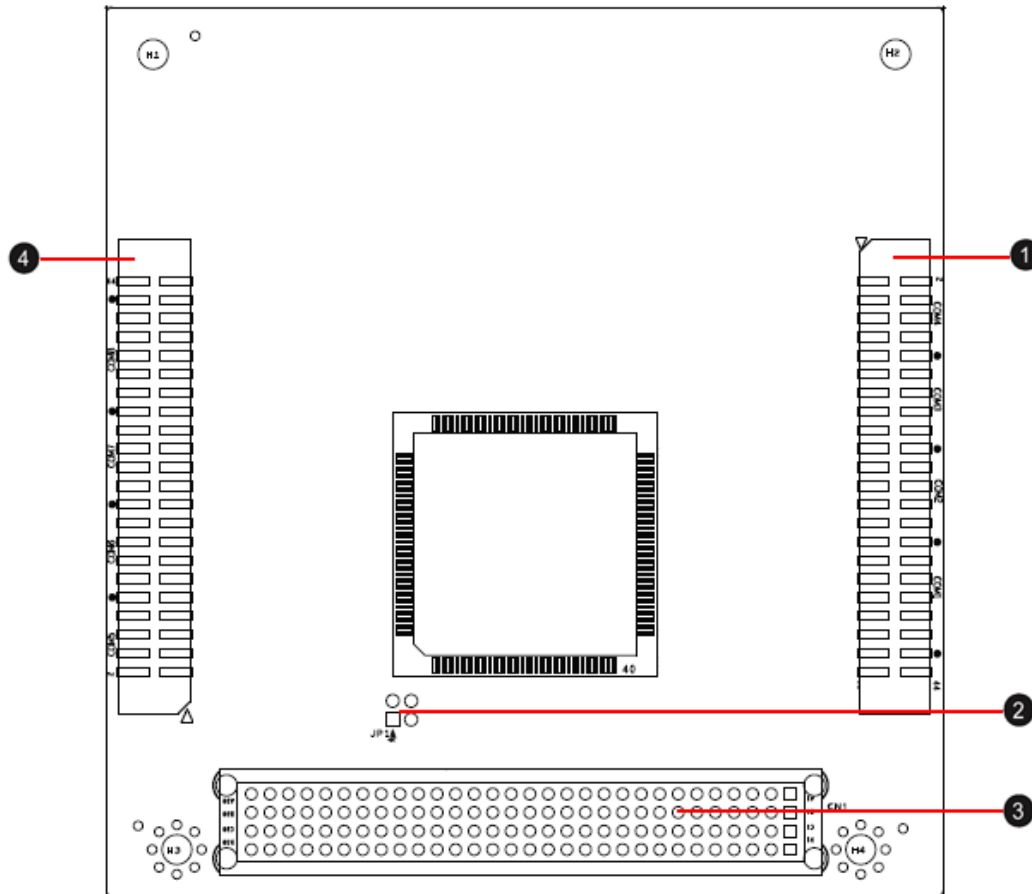
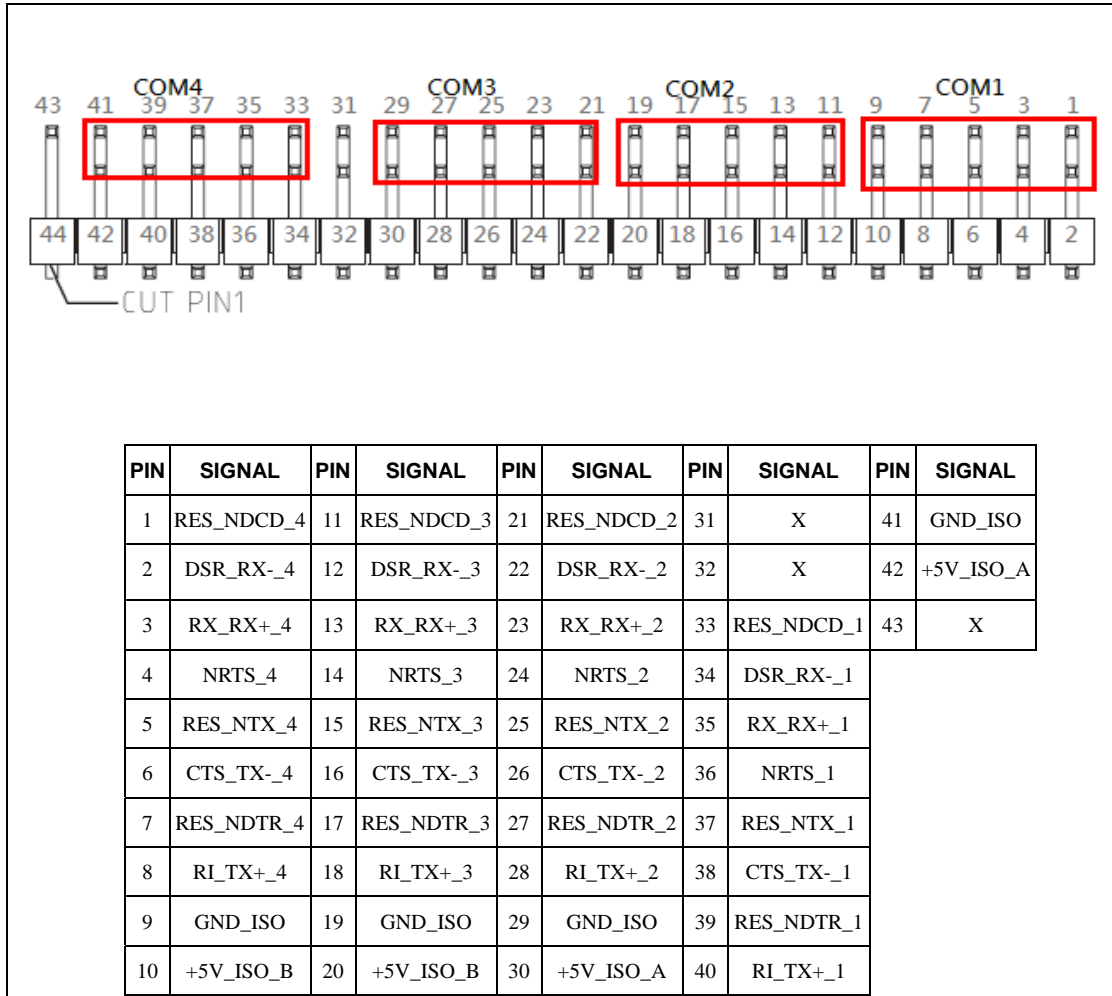


Figure 2: Locations (Top side)

1	<b>COM_1_4</b> COM 1 TO COM 4
2	<b>JP1</b> DEVICE SELECT
3	<b>CN1</b> PCI104 CONNECTOR
4	<b>COM_5_8 (WITHOUT AR-B104B_4P)</b> COM5 TO COM8
	<b>COM1~COM8</b> DB-9 Serial port pin assignment

## 2.2 Connectors and Jumper Setting

### 2.2.1 COM\_1\_4



### 2.2.2 JP1



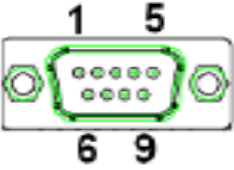
### 2.2.3 COM\_5\_8

43	41	39	37	35	33	31	29	27	25	23	21	19	17	15	13	11	9	7	5	3	1
44	42	40	38	36	34	32	30	28	26	24	22	20	18	16	14	12	10	8	6	4	2
COM8				COM7				COM6				COM5									

PIN	SIGNAL	PIN	SIGNAL	PIN	SIGNAL	PIN	SIGNAL	PIN	SIGNAL
1	RES_NDCD_5	11	RES_NDCD_6	21	RES_NDCD_7	31	X	41	GND_ISO
2	DSR_RX-_5	12	DSR_RX-_6	22	DSR_RX-_7	32	X	42	+5V_ISO_D
3	RX_RX+_5	13	RX_RX+_6	23	RX_RX+_7	33	RES_NDCD_8	43	X
4	NRTS_5	14	NRTS_6	24	NRTS_7	34	DSR_RX-_8	44	X
5	RES_NTX_5	15	RES_NTX_6	25	RES_NTX_7	35	RX_RX+_8		
6	CTS_TX-_5	16	CTS_TX-_6	26	CTS_TX-_7	36	NRTS_8		
7	RES_NDTR_5	17	RES_NDTR_6	27	RES_NDTR_7	37	RES_NTX_8		
8	RI_TX+_5	18	RI_TX+_6	28	RI_TX+_7	38	CTS_TX-_8		
9	GND_ISO	19	GND_ISO	29	GND_ISO	39	RES_NDTR_8		
10	+5V_ISO_C	20	+5V_ISO_C	30	+5V_ISO_D	40	RI_TX+_8		

### 2.2.4 COM1~COM8 : DB-9 Serial port pin assignment

				
PIN	RS232	RS422	RS485	
1	-DCD	TX-	DATA-	
2	RX	TX+	DATA+	
3	TX	RX-		
4	-DTR	RX+		
5	GND	GND	GND	
6	-DSR			
7	-RTS			
8	-CTS			
9	-RI			

## 3 BIOS SETTING

This chapter describes the BIOS menu displays and explains how to perform common tasks needed to get up and running. It also gives detailed explanation of the elements found in each of the BIOS menus. The following topics are covered:

- AR-B104B Selectable Setup

### 3.1 AR-B104B Selectable Setup

Once you enter the Award BIOS™ CMOS Setup Utility, the Main Menu will appear on the screen. Use the arrow keys to highlight the item and then use the <Pg Up> <Pg Dn> keys to select the value. You want in each item.

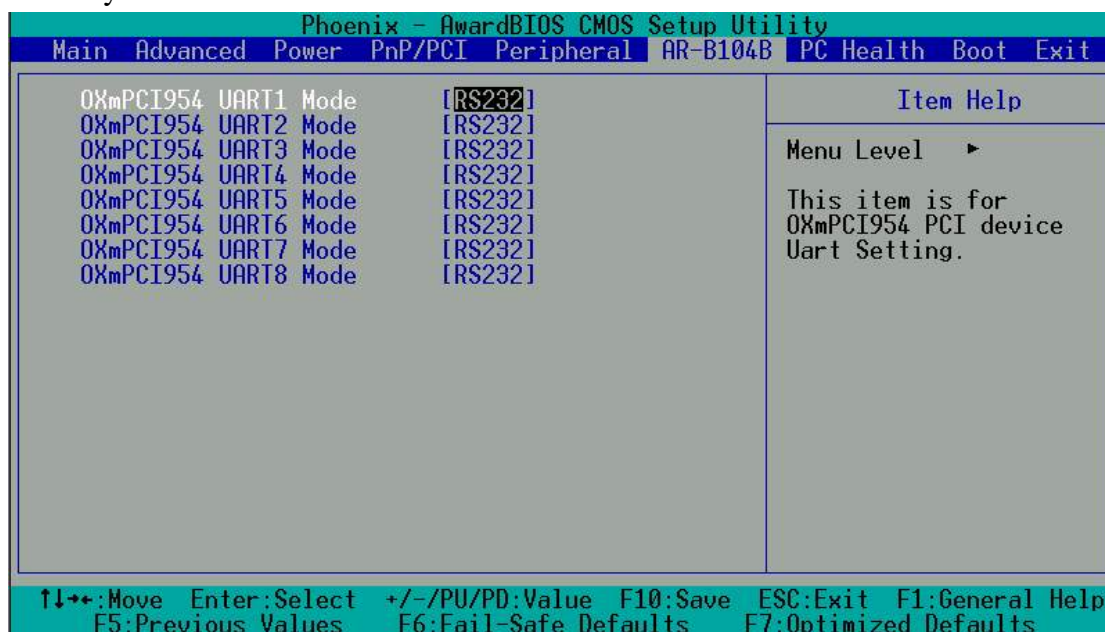


Figure 4: AR-B104B Selectable setup

Note : Listed at the bottom of the menu are the control keys. If you need any help with the item fields, you can press the <F1> key, and it will display the relevant information.

Option	Choice	Description
OxmPCI954 UART(X) Mode	RS232 RS422 RS485	This category select RS232, RS422, RS485 mode

# 4 SOFTWARE

## 4.1 Constant Definition

```
/*Include Lib or Dll Functions */
#include "ARB104B_LIB.h"
#define TRUE 1
#define FALSE 0
/*8 COM Port */
#define M8_232 0x00DB6DB6
#define M8_422 0x00249249
#define M8_485 0x006DB6DB
/*Serial COM*/
#define M_232 0x06
#define M_422 0x01
#define M_485 0x03
/*Input Data shift numbers*/
#define com1_offset 21
#define com2_offset 18
#define com3_offset 15
#define com4_offset 12
#define com5_offset 9
#define com6_offset 6
#define com7_offset 3
#define com8_offset 0
```

## 4.2 API Definition & Description

### □ unsigned int WMODE(unsigned long DATA)

Write Data Format :

**TABLE 1. Data Mode Format**

31									0
N/A	N/A	COM1	COM2	COM3	COM4	COM5	COM6	COM7	COM8
xxxx	xxxx	000	000	000	000	000	000	000	000

Argument = 32 bit DATA

Return Value:

1 : Write Error.

0 : Write Success.

### □ unsigned int checkmode(void)

Read 4Port or 8Port COM functions

Return Value:

4 : 4 COM Port.

8 : 8 COM Port.

1 : Error.

### 4.3 Sample Code :

#### 4.3.1 Set Mode

```
unsigned long  com1,com2,com3,com4,com5,com6,com7,com8;//Set COM Value
unsigned long commode=0;                                // Set Output COM
port Value

com1 = M_232 << com1_offset;  //Set COM1 = RS232 Mode
com2 = M_232 << com2_offset;  //Set COM2 = RS232 Mode
com3 = M_422 << com3_offset;  //Set COM3 = RS422 Mode
com4 = M_485 << com4_offset;  //Set COM4 = RS485 Mode
com5 = M_422 << com5_offset;  //Set COM5 = RS422 Mode
com6 = M_232 << com6_offset;  //Set COM6 = RS232 Mode
com7 = M_485 << com7_offset;  //Set COM7 = RS485 Mode
com8 = M_422 << com8_offset;  //Set COM8 = RS422 Mode

//Output Data based on COM1 to COM4
commode = (com1 | com2 | com3 | com4) & 0x00FFF000;

//Output Data based on COM1 to COM8
commode = (com1 | com2 | com3 | com4 | com5 | com6 | com7 | com8)&0x00FFFFFF;
error = WMODE(commode);  //Write Mode Data
if(error !=0)             //if error = 1,  Write Error
    printf("    Write error\n");
else
    printf("    Write ok\n");  //if error=0,  Write Ok
```

#### 4.3.2 Read 4P/8P COM Port Mode

```
unsigned int mode;      // Set COM Port Mode Value
mode = checkmode();    // Read mode Data
if(mode == 1)          //if mode = 1,  error,
    printf(" Error \n");
else
    printf("mode = %d",mode); //if mode = 4,  COM Port =4,  mode = 8,
COM Port=8.
```



## 4.4 The Function Detected 8 PORT

### 4.4.1 Edit boot

```
#vim /boot/grub/grub.conf
```

Exesample:

```
[root@linux ~]# vi /boot/grub/grub.conf
```

```
default=0
```

```
timeout=5
```

```
splashimage=(hd0,0)/grub/splash.xpm.gz
```

```
hiddenmenu
```

```
title Fedora Core (2.6.27.5.117.FC10)
```

```
    root (hd0,0)
```

```
    kernel /vmlinuz-2.6.27.5.117.FC10 ro root=/dev/hda2 rhgb quiet
```

```
    8250.nr_uarts=12 noirqdebug
```

```
    initrd /initrd-2.6.27.5.117.FC10.img
```

Add “quiet” after **8250.nr\_uarts=12 noirqdebug**

**“12” is the number of supported port.**

### 4.4.2 Reset

To execute the code, to see the devices connected.

```
#setserial -g /dev/ttyS* reports
```

```
/dev/ttySx, UART: 16950/954, Port: 0xxxxx, IRQ: x
```

```
/dev/ttySx, UART: 16950/954, Port: 0xxxxx, IRQ: x
```

```
/dev/ttySx, UART: 16950/954, Port: 0xxxxx, IRQ: x
```

```
/dev/ttySx, UART: 16950/954, Port: 0xxxxx, IRQ: x
```

```
/dev/ttySx, UART: 16950/954, Port: 0xxxxx, IRQ: x
```

```
/dev/ttySx, UART: 16950/954, Port: 0xxxxx, IRQ: x
```

```
/dev/ttySx, UART: 16950/954, Port: 0xxxxx, IRQ: x
```

```
/dev/ttySx, UART: 16950/954, Port: 0xxxxx, IRQ: x
```

**\* This method confirmed only on “FC10”**